

Öğrenci Adı Soyadı : Hüseyin Karakaya

Öğrenci Numarası: 170422022

T.C. MARMARA ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ
2024-2025 EĞİTİM ÖĞRETİM YILI
GÜZ DÖNEMİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİTABANI YÖNETİM SİSTEMLERİ DERSİ
PROJE-FİNAL DÖKÜMANI

1. Gerçekleştirdiğiniz veri tabanı projesi için grup arkadaşlarınızın isimlerini yazınız ve projenize ait veri tabanı/diğer yazılım bileşenleri hakkında bilgi veriniz. (5 p)

Grup çalışmamızda Burak Akça ve Ozan Deste ile birlikte çalışarak bir Araç Bakım ve Servis Takip Sistemi geliştirdik. Bu proje, araçların bakım süreçlerini ve servis kayıtlarını daha etkili bir şekilde yönetmek için tasarlandı. Projemiz, masaüstü bir uygulama olarak **JavaFX** kullanılarak geliştirildi. Arka planda, veri yönetimi için **Microsoft SQL Server (MSSQL)** tercih edildi.

2. Gerçekleştirdiğiniz veri tabanı projesi için proje dokümanınızı ve dosyalarınızı içeren herkese açık github bağlantılarınızı paylaşınız. (5 p)

<https://github.com/karakayahuseyin/arac-servis-bakim>

3. Gerçekleştirdiğiniz projenin amacını detaylı bir şekilde açıklayınız. (10 p)

Bu proje, bir araç servis işletmesinin yedek parça ve servis operasyonlarını verimli bir şekilde takip edebilmesi ve yönetebilmesi için bir veritabanı tasarımı ve uygulamasını amaçlamaktadır. Servis süreçlerinin kayıt altına alınması, yedek parça stoklarının yönetimi ve müşterilerle ilgili detaylı bilgilerin saklanması için bir çözüm sunar. Projenin amaçlarını detaylandırıcak olursak:

Servis Süreçlerini Takip Etmek:

- Araçların muayene ve tamir süreçleriyle ilgili tüm bilgileri bir arada tutarak düzenli bir kayıt sistemi oluşturmak.
- Servis teknisyenlerinin hangi muayenelerde görev aldığını ve tamir işlemlerinde kullanılan işçilik ve yedek parça bilgilerini saklamak.
- Hangi araçların hangi tarihlerde servise geldiğini ve tamir masraflarını takip etmek.

Yedek Parça Stok Yönetimi:

- Araç tamirlerinde kullanılan yedek parçaların stok durumlarını ve maliyetlerini kontrol altında tutmak.
- Hangi yedek parçaların hangi araç modellerinde kullanıldığını kaydetmek.
- Yedek parça stoklarında eksikliklerin önüne geçerek zamanında tedarik edilmesini sağlamak.

Müşteri ve Araç Bilgilerinin Yönetimi:

- Müşterilerin kişisel bilgilerini (isim, soyisim, iletişim bilgileri vb.) ve sahip oldukları araçları kayıt altına almak.
- Müşterilerin hangi araçlarıyla ne zaman servise geldiklerini ve toplamda ne kadar masraf yaptıklarını görmek.
- Müşteri odaklı bir veri yönetimi sağlayarak müşteri memnuniyetini artırmak.

4. Tasarladığınız veri tabanı mimarisinde hangi tablo ve ilişkileri kullandığınızı açıklayınız. (10 p)

Tablolar

1-Müşteri:

- Müşterilerin bilgilerini tutar.
- Alanlar: id, Ad, Soyad, Mail, Yaş, Password, Kullanıcı_Adı.

2-MüşteriAraba:

- Müşterilere ait araç bilgilerini içerir.
- Alanlar: Plaka, Araba_Modeli, Ruhsat_Sahibi.

3-ArabaModeli:

- Araç modelleri ile ilgili bilgileri barındırır.
- Alanlar: id, Marka, Modeli, Üretim_Yılı, Periyodik_Bakım_Süresi.

4-YedekParça:

- Araç yedek parçalarını içerir.
- Alanlar: id, Yedek_Parça_Adı, Stok, Fiyat, Araba_Modeli.

5-Muayene:

- Araçların muayene kayıtlarını tutar.
- Alanlar: id, Servis_Teknikeri, Tarih, Plaka.

6-Tamir:

- Araç tamir bilgilerini barındırır.
- Alanlar: id, Yedek_Parça, Tamir_Süresi, İşçilik_Ücreti, Toplam_Ücret, Muayene_id.

7-ServisTeknikeri:

- Servis teknikerlerinin bilgilerini tutar.
- Alanlar: id, Ad, Soyad, Mail, Password, Kullanıcı_Adı, Yaş.

İlişkiler

1-Müşteri - MüşteriAraba:

- Bir müşteri birden fazla araca sahip olabilir. Ruhsat_Sahibi alanı ile ilişki kuruldu.

2-MüşteriAraba - ArabaModeli:

- Her aracın modeli vardır. Araba_Modeli alanı bu ilişkiyi ifade eder. Bir araba modeli birden fazla kullanıcıda olabilir.

3-ArabaModeli - YedekParça:

- Her araç modeline uygun yedek parçalar vardır. Bir arabada birden çok yedek parça kullanılabilir.

4-Muayene - ServisTeknikeri:

- Muayene işlemi bir tekniker tarafından gerçekleştirilir. Bir servis teknikeri birden fazla muayene yapabilir

5-Muayene - MüşteriAraba:

- Muayene edilen araçlar Plaka alanı üzerinden ilişkilendirildi. Bir araba birden fazla muayenede yer alabilir

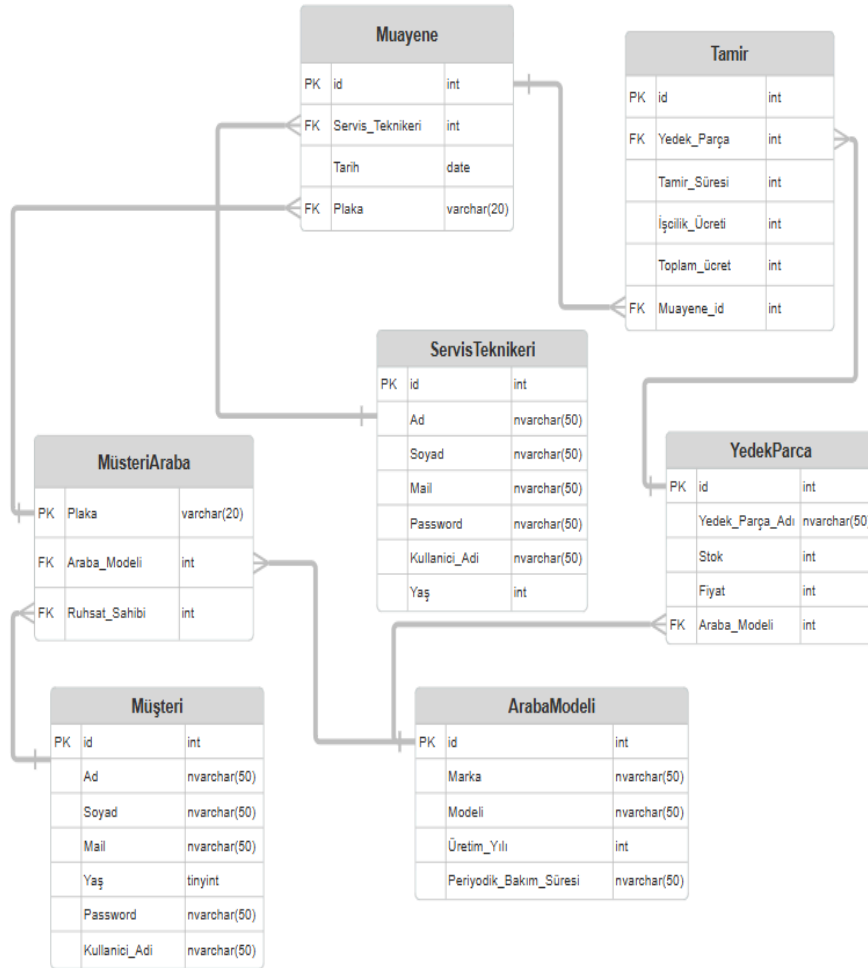
6-Tamir - Muayene:

- Tamir işlemleri bir muayeneye bağlıdır. Muayene_id ile ilişki kuruldu. Bir muayenede birden fazla tamir yer alabilir

7-Tamir - YedekParça:

Tamir işlemlerinde kullanılan yedek parçalar Yedek_Parça alanı üzerinden ilişkilendirilmiştir. Bir yedek_parça birden fazla tamirde kullanılabilir.

5. Veri tabanı ER (Entity Relationship) diagramının bilgisayar ortamında çizilmiş halini paylaşınız. (Ara raporda eksik kısımlar bu raporda giderilmelidir ve ER çizme programlarından faydalanılabilir. Elle çizim, çizip fotoğrafını çekme vb. kabul edilmeyecektir.) (10 p)



6. Herhangi iki tablonuz için DDL (create) kodları yazılmalıdır. (10 p)

Musteri Tablosu

```
CREATE TABLE Musteri ( id INT PRIMARY KEY IDENTITY(1,1),  
Ad NVARCHAR(50),  
Soyad NVARCHAR(50) ,  
Mail NVARCHAR(50),  
Yas tinyint,  
Password NVARCHAR(50) ,  
Kullanici_Adi NVARCHAR(50) );
```

MusteriAraba Tablosu

```
CREATE TABLE MusteriAraba (  
Plaka VARCHAR(20) PRIMARY KEY,  
Araba_Modeli INT,  
Ruhsat_Sahibi INT,  
FOREIGN KEY (Araba_Modeli) REFERENCES ArabaModeli(id),  
FOREIGN KEY (Ruhsat_Sahibi) REFERENCES Musteri(id));
```

7. 5 adet DML (update, insert, delete) içeren kodları yazılmalıdır. (10 p)

1. Yeni Müşteri Ekleme (INSERT)

```
INSERT INTO Musteri (Ad, Soyad, Mail, Yaş, Password, Kullanici_Adi)  
VALUES ('Ahmet', 'Yılmaz', 'ahmet.yilmaz@mail.com', 35, '12345', 'ahmet35');
```

2. Yeni Araç Ekleme (INSERT)

```
INSERT INTO MüsteriAraba (Plaka, Araba_Modeli, Ruhsat_Sahibi)  
VALUES ('07p2316', 40, 4);
```

3. Araç Modelinin Periyodik Bakım Süresini Güncelleme (UPDATE)

UPDATE ArabaModeli

SET Periyodik_Bakım_Süresi = '12 ay'

WHERE id = 40;

4. Bir Tamir Kaydını Silme (DELETE)

DELETE FROM Tamir

WHERE id = 2;

5. Müşteri Bilgilerini Güncelleme (UPDATE)

UPDATE Musteri SET Mail = 'mehmet.kaya@mail.com', Yaş = 36 WHERE id = 4;

8. Projenize ait kendi belirlediğiniz 10 adet SQL sorgusu yazınız, sorguların amacını ve sonuç çıktısını da lütfen ekleyiniz. (Açıklama: Sorgular Select deyimleri ve gruptama fonksiyonlarını HAVING deyimini (min, max, avg, count gibi) ve join deyimlerini (en az iki tablo ile birleştirme sorgusu) içerecek şekilde basitten karmaşığa doğru gitmelidir. Proje sunum anında veri tabanınıza ait sorular SQL ortamında gösterilecek ve açıklanacaktır. Raporunuzda ise sorgular, sorguların cevap ve sonuçlarının ekran görüntüsü olarak paylaşılması beklenmektedir. (10 p) Örnek:

1. Tüm Müşterileri Listeleme

Amaç: Sistemde kayıtlı tüm müşterilerin adını, soyadını ve e-posta adreslerini listelemek.

SELECT Ad, Soyad, Mail

FROM Musteri;

SELECT Ad, Soyad, Mail			
FROM Musteri;			
109 %			
Results Messages			
	Ad	Soyad	Mail
1	Ali	Kaya	mehmet.kaya@mail.com
2	Ayşe	Yılmaz	ayse.yilmaz@mail.com
3	Mehmet	Demir	mehmet.demir@mail.com
4	Fatma	Çelik	fatma.celik@mail.com
5	Can	Arslan	can.arslan@mail.com
6	Elif	Şahin	elif.sahin@mail.com
7	Osman	Kurt	osman.kurt@mail.com
8	Leyla	Çakır	leyla.cakir@mail.com
9	Murat	Öztürk	murat.ozturk@mail.com
10	Zeynep	Aydın	zeynep.aydin@mail.com
11	Burak	Akça	-
12	Burak	Akça	-
13	Ahmet	Yılmaz	ahmet.yilmaz@mail.com
14	Ad1	Soyad1	user1@mail.com
15	Ad2	Soyad2	user2@mail.com
16	Ad3	Soyad3	user3@mail.com
17	Ad4	Soyad4	user4@mail.com
18	Ad5	Soyad5	user5@mail.com
19	Ad6	Soyad6	user6@mail.com
20	Ad7	Soyad7	user7@mail.com
21	Ad8	Soyad8	user8@mail.com
22	Ad9	Soyad9	user9@mail.com
23	Ad10	Soya...	user10@mail.com

2. En Çok Stokta Olan Yedek Parçaları Listeleme

Amaç: Stokta bulunan yedek parçaları azalan stok miktarına göre listeleme.

```
SELECT Yedek_Parça_Adı, Stok
```

```
FROM YedekParca
```

```
ORDER BY Stok DESC;
```

SQLQuery3.sql - loc...visi1 (MSI\msi (67))* × MSI\SQL

```

SELECT Yedek_Parça_Adı, Stok
FROM YedekParca
ORDER BY Stok DESC;

```

109 %

Results Messages

	Yedek_Parça_Adı	Stok
1	Buji	70
2	Amortisör	60
3	Fren Balatası	50
4	Yağ Filtresi	30
5	Debriyaj Seti	20

3. Araba Modellerine Göre Yedek Parça Sayısı

Amaç: Her araba modeli için kaç farklı yedek parça bulunduğunu listeleme.

```

SELECT Araba_Modeli, COUNT(*) AS Parca_Sayisi
FROM YedekParca
GROUP BY Araba_Modeli;

```

SQLQuery3.sql - loc...visi1 (MSI\msi (67))* × MSI\SQL EXPRESS.A...- db

```

SELECT Araba_Modeli, COUNT(*) AS Parca_Sayisi
FROM YedekParca
GROUP BY Araba_Modeli;

```

109 %

Results Messages

	Araba_Modeli	Parca_Sayisi
1	35	1
2	36	1
3	37	1
4	38	1
5	40	1

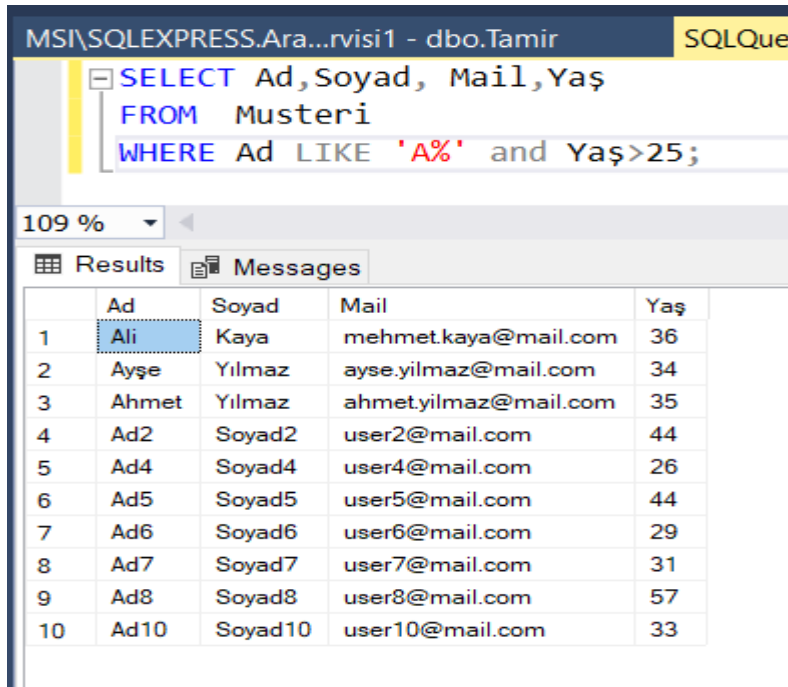
4. Verilen Koşula Göre Müşteri Bilgilerini Getirme

Amaç: İsmi A ile başlayan ve yaşı 25 den büyük olan müşteri bilgileri listeleme.

```
SELECT Ad,Soyad, Mail,Yaş
```

```
FROM Musteri
```

```
WHERE Ad LIKE 'A%' and Yaş>25
```



The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays the following SQL query:

```
SELECT Ad,Soyad, Mail,Yaş
FROM Musteri
WHERE Ad LIKE 'A%' and Yaş>25;
```

The bottom pane shows the results of the query in a table format. The table has five columns: Ad, Soyad, Mail, and Yaş. The results are as follows:

	Ad	Soyad	Mail	Yaş
1	Ali	Kaya	mehmet.kaya@mail.com	36
2	Ayşe	Yılmaz	ayse.yilmaz@mail.com	34
3	Ahmet	Yılmaz	ahmet.yilmaz@mail.com	35
4	Ad2	Soyad2	user2@mail.com	44
5	Ad4	Soyad4	user4@mail.com	26
6	Ad5	Soyad5	user5@mail.com	44
7	Ad6	Soyad6	user6@mail.com	29
8	Ad7	Soyad7	user7@mail.com	31
9	Ad8	Soyad8	user8@mail.com	57
10	Ad10	Soyad10	user10@mail.com	33

5. 2023 Yılında Yapılan Muayeneler

Amaç: 2023 yılında yapılan tüm muayeneleri listeleme.

```
SELECT *
```

```
FROM Muayene
```

WHERE YEAR(Tarih) = 2023;

MSI\SQLEXPRESS.Ara...rvisi1 - dbo.Tamir SQLQu

```
SELECT *  
FROM Muayene  
WHERE YEAR(Tarih) = 2023;
```

109 %

Results Messages

	id	Servis_Teknikeri	Tarih	Plaka
1	14	7	2023-08-25	38PTR534

6. Bir Plakaya Ait Müşteri Bilgilerini Getirme

Amaç: Belirli bir plakaya ait müşteri bilgilerini getirme.

SELECT M.Ad, M.Soyad, MA.Plaka

FROM Musteri M

JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi

WHERE MA.Plaka = '07p2316';

MSI\SQLEXPRESS.Ara...rvisi1 - dbo.Tamir SQLQuery3.sql - loc...visi1 (MSI\ms

```
SELECT M.Ad, M.Soyad, MA.Plaka  
FROM Musteri M  
JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi  
WHERE MA.Plaka = '07p2316';
```

109 %

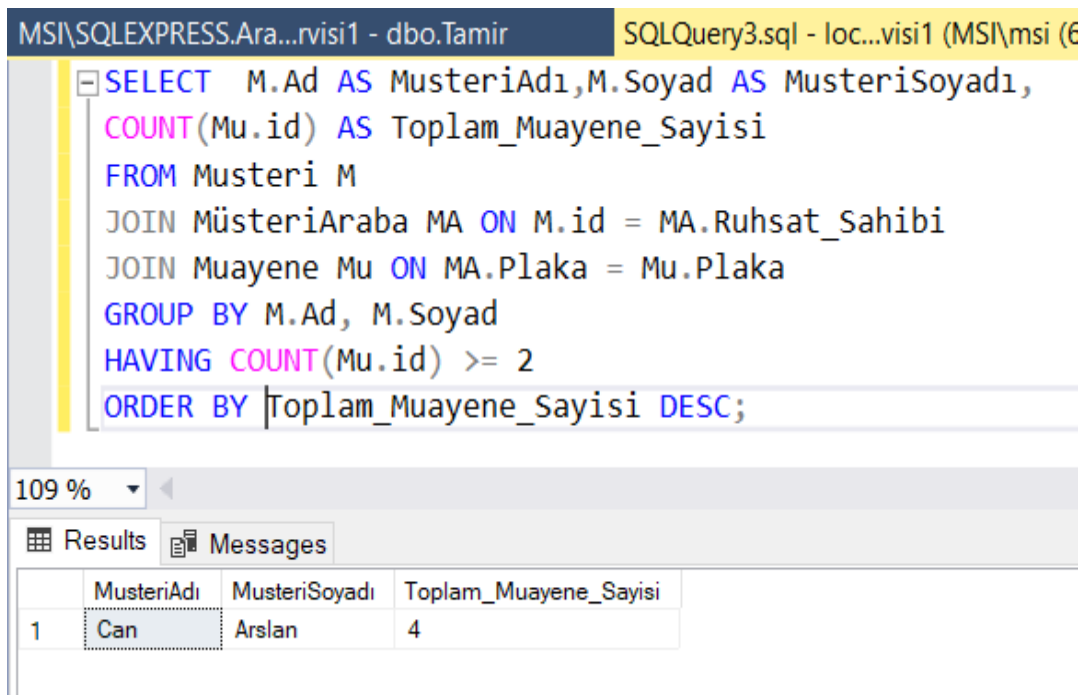
Results Messages

	Ad	Soyad	Plaka
1	Ali	Kaya	07p2316

7. En Az İki Muayeneye Sahip Müşteriler

Amaç: Her müşterinin sahip olduğu araçlar için toplam muayene sayısını hesapla ve en az 2 muayenesi olan müşterileri listeleme.

```
SELECT M.Ad AS MusteriAdı,M.Soyad AS MusteriSoyadı,  
COUNT(Mu.id) AS Toplam_Muayene_Sayisi  
FROM Musteri M  
JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi  
JOIN Muayene Mu ON MA.Plaka = Mu.Plaka  
GROUP BY M.Ad, M.Soyad  
HAVING COUNT(Mu.id) >= 2  
ORDER BY Toplam_Muayene_Sayisi DESC;
```



The screenshot shows a SQL Server Enterprise Manager window. The top bar indicates the connection is 'MSI\SQLEXPRESS.Ara...rvisi1 - dbo.Tamir' and the file is 'SQLQuery3.sql - loc...visi1 (MSI\msi (6...'. The main area displays the following SQL query:

```
SELECT M.Ad AS MusteriAdı,M.Soyad AS MusteriSoyadı,  
COUNT(Mu.id) AS Toplam_Muayene_Sayisi  
FROM Musteri M  
JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi  
JOIN Muayene Mu ON MA.Plaka = Mu.Plaka  
GROUP BY M.Ad, M.Soyad  
HAVING COUNT(Mu.id) >= 2  
ORDER BY Toplam_Muayene_Sayisi DESC;
```

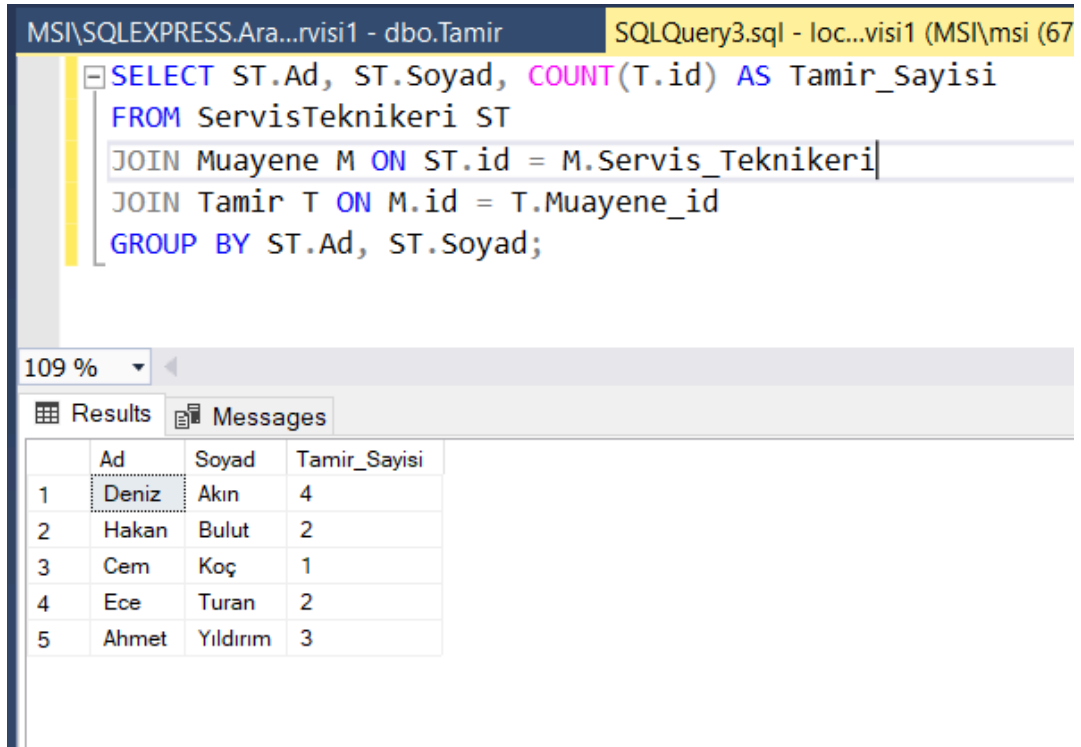
Below the query editor, the 'Results' tab is active, showing a table with the following data:

	MusteriAdı	MusteriSoyadı	Toplam_Muayene_Sayisi
1	Can	Arslan	4

8. Her Servis Teknisyeninin Tamir Sayısı

Amaç: Her servis teknisyeninin kaç tamir işlemi yaptığını bulma.

```
SELECT ST.Ad, ST.Soyad, COUNT(T.id) AS Tamir_Sayisi  
FROM ServisTeknikeri ST  
JOIN Muayene M ON ST.id = M.Servis_Teknikeri  
JOIN Tamir T ON M.id = T.Muayene_id  
GROUP BY ST.Ad, ST.Soyad;
```



The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results grid. The query editor contains the following SQL code:

```
SELECT ST.Ad, ST.Soyad, COUNT(T.id) AS Tamir_Sayisi  
FROM ServisTeknikeri ST  
JOIN Muayene M ON ST.id = M.Servis_Teknikeri  
JOIN Tamir T ON M.id = T.Muayene_id  
GROUP BY ST.Ad, ST.Soyad;
```

The results grid displays the following data:

	Ad	Soyad	Tamir_Sayisi
1	Deniz	Akın	4
2	Hakan	Bulut	2
3	Cem	Koç	1
4	Ece	Turan	2
5	Ahmet	Yıldırım	3

9. Müşterilerin Sahip Olduğu Araçlara Göre Toplam Muayene Sayısı

Amaç: Bu sorgu, her bir müşterinin sahip olduğu araçlarla ilgili toplam muayene sayısını hesaplar.

```
SELECT M.Ad AS MusteriAdı,M.Soyad AS MusteriSoyadı,COUNT(Mu.id) AS  
Toplam_Muayene_Sayisi
```

```
FROM Musteri M
```

```
JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi
```

```
JOIN Muayene Mu ON MA.Plaka = Mu.Plaka
```

```
GROUP BY M.Ad, M.Soyad
```

```
ORDER BY Toplam_Muayene_Sayisi DESC;
```

The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results grid. The query editor displays the following SQL query:

```
SELECT M.Ad AS MusteriAdı,M.Soyad AS MusteriSoyadı,COUNT(Mu.id)  
AS Toplam_Muayene_Sayisi  
FROM Musteri M  
JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi  
JOIN Muayene Mu ON MA.Plaka = Mu.Plaka  
GROUP BY M.Ad, M.Soyad  
ORDER BY Toplam_Muayene_Sayisi DESC;
```

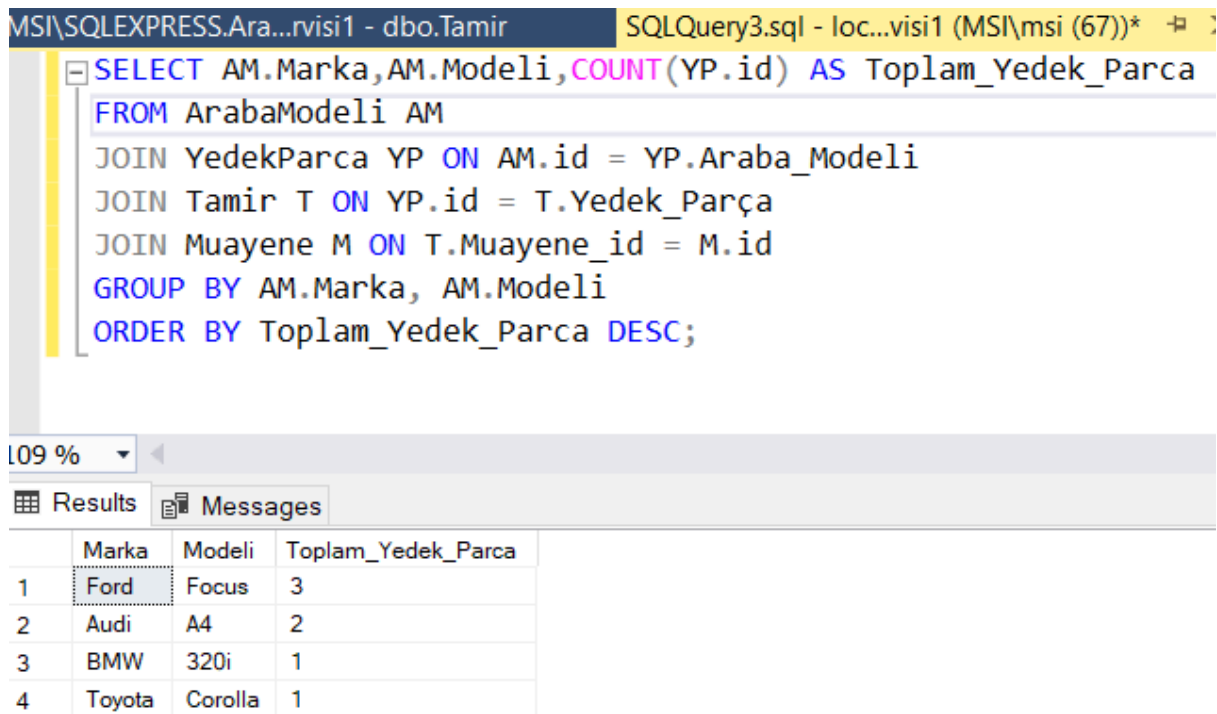
The results grid shows the following data:

	MusteriAdı	MusteriSoyadı	Toplam_Muayene_Sayisi
1	Can	Arslan	4
2	Zeynep	Aydın	1
3	Leyla	Çakır	1
4	Fatma	Çelik	1
5	Mehmet	Demir	1
6	Ali	Kaya	1
7	Osman	Kurt	1
8	Murat	Öztürk	1
9	Elif	Şahin	1
10	Ayşe	Yılmaz	1

10. En Fazla Yedek Parça Kullanan Araç Modellerini Bulma

Amaç: Bu sorgu, araç modellerine göre kullanılan toplam yedek parça sayısını hesaplar ve sıralar.

```
SELECT AM.Marka,AM.Modeli,COUNT(YP.id) AS Toplam_Yedek_Parça
FROM ArabaModeli AM
JOIN YedekParca YP ON AM.id = YP.Araba_Modeli
JOIN Tamir T ON YP.id = T.Yedek_Parça
JOIN Muayene M ON T.Muayene_id = M.id
GROUP BY AM.Marka, AM.Modeli
ORDER BY Toplam_Yedek_Parça DESC;
```



The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results grid. The query editor contains the following SQL code:

```
SELECT AM.Marka,AM.Modeli,COUNT(YP.id) AS Toplam_Yedek_Parça
FROM ArabaModeli AM
JOIN YedekParca YP ON AM.id = YP.Araba_Modeli
JOIN Tamir T ON YP.id = T.Yedek_Parça
JOIN Muayene M ON T.Muayene_id = M.id
GROUP BY AM.Marka, AM.Modeli
ORDER BY Toplam_Yedek_Parça DESC;
```

The results grid displays the following data:

	Marka	Modeli	Toplam_Yedek_Parça
1	Ford	Focus	3
2	Audi	A4	2
3	BMW	320i	1
4	Toyota	Corolla	1

9. Eğer gerçekleştirmiş iseniz, veri tabanı bağlama ve uygulama geliştirme aşamalarınızı kısaca açıklayarak, kullanıcı ara yüz ekranından bir örnek veriniz. Ve geliştirdiğiniz ara yüzü anlatınız. (10 p)

- a) SQL Server Configuration Manager üzerinde SQL Server Hizmetleri aktifleştirildi.
- b) SQL Server Configuration Manager üzerinde SQL Server Network Configuration bölümünden TCP/IP etkinleştirildi ve çalıştırıldığı port adresi bulundu. (Projemiz için 1433 olacaktır)
- c) MSSQL'e ait JDBC driver dosyaları indirildi ve indirilen driver dosyasındaki 12.8.1 sürümüne ait jar dosyası projeye kütüphane olarak eklendi.
- d) module-info.java içerisinde SQL kütüphanesi import edildi module ... { requires java.sql; }
- e) Veritabanına bağlantıyı sağlamak için controller/VeritabaniBaglantisi sınıfı oluşturuldu.

```
public class VeritabaniBaglantisi {
```

```
    private static Connection connection;
```

```
    private static final String URL = "jdbc:sqlserver...";
```

```
    private static final String USERNAME = "...";
```

```
    private static final String PASSWORD = "...";
```

```
    private static Connection getConnection() throws SQLException {
```

```
        if(connection == null || connection.isClosed) {
```

```
            connection = DriverManager.getConnection(URL, USERNAME, PASSWORD);
```

```
            System.out.println("Veritabanına bağlanıldı.");
```

```
        }
```

```
        return connection;}}
```

- f) Hem müşteri kullanıcı hem servis teknikeri kullanıcısı için yetkilere göre giriş sağlandı.

- g) Arayüz üzerinde görüntülenen tabloların her biri için model sınıfları oluşturuldu

Örn:

AraclarımTabloSatır(String markaModel, String plaka, String tamirSuresi, String periyodikBakimSuresi)

ArizaTamirSatır(int yedekParçaId, String yedekParca, int tamirSuresi, int iscilikUcreti, int toplamUcret)

Örn Servis Teknikeri ekranı üzerindeki Stok İşlemleri Ekranı:

[illegible]

Yedek Parça bilgilerini görüntüleyen tablo ve ekleme, silme ve düzenleme ekranlarını açan butonlar yer almaktadır. Düzenle ekranı seçilen satır ile birlikte tıklandığında:

[illegible]

Örn Müşteri ekranı:

The screenshot shows a web application window titled "Müşteri İşlemleri". Below the title bar, there's a section "ARAÇ SAHİBİ İŞLEMLERİ". Under this, there's a sub-section "ARAÇLARIM" with a button "Araç Detayı Görüntüle". Below this is a table with 4 columns: "Marka/ Model", "Plaka", "Durum", and "Sonraki Periyodik Bakım". The table contains 3 rows of vehicle data. Below the table, there are two sections: "Muayene Geçmişi" and "Tamir Geçmişi", each with its own table. The "Muayene Geçmişi" table has 2 columns: "Tarih" and "Arıza/Tamir Durumu". The "Tamir Geçmişi" table has 6 columns: "Arıza", "Süre (Saat)", "Baş. Tarihi", "Marka", and "Model".

Marka/ Model	Plaka	Durum	Sonraki Periyodik Bakım
Mercedes C200	07MNO234	Hazır	24 ay
Audi A4	38PTR534	Hazır	24 ay
Honda Civic	44FZR534	Tamirde	12 ay

Tarih	Arıza/Tamir Durumu
2024-02-...	44FZR534
2024-01-...	44FZR534
2024-12-...	44FZR534
2024-12-...	44FZR534
2024-12-...	44FZR534
2024-12-...	44FZR534
2024-10-...	44FZR534
2014-12-...	44FZR534

Arıza	Süre (Saat)	Baş. Tarihi	Marka	Model
Fren Balatası	3	2024-10-10	Honda	Civic
Yağ Filtresi	4	2024-10-10	Honda	Civic
Direksiyon ...	4	2014-12-12	Honda	Civic
Yağ Filtresi	4	2014-12-12	Honda	Civic

Bu arayüz üzerinde müşteri kendi araçlarına ait genel bilgileri ve istediği takdirde tablodan seçeceği aracın geçmiş muayene ve tamir detay bilgilerine erişebilmektedir.

10. Eğer veri tabanı bağlama işlemini gerçekleştirmemiş iseniz VTYS sistemlerinde Transaction nedir açıklayınız ve çalışmanızdan bir Transaction örneği veriniz. (10 p)

11. View nedir açıklayınız ve bir adet view, bir adet saklı yordam (Stored Procedure) ifadesine ait SQL deyimlerinin sorgusunu ve cevabını yazınız. (10 p)

View (Görünüm), veritabanındaki bir veya birden fazla tablodan oluşturulan sanal bir tablodur. View, fiziksel olarak veri içermez, yalnızca verilerin bir sorgu sonucunda nasıl görüneceğini tanımlar. Genellikle karmaşık sorguları basitleştirmek, veri güvenliğini artırmak veya belirli verileri kolayca erişilebilir hale getirmek için kullanılır.

Stored Procedure (Saklı Yordam) ise birden fazla SQL deyimini içerebilen ve belirli bir görevi yerine getiren, veritabanında saklanan ve gerektiğinde çağrılabilen bir program birimidir. Performansı artırmak ve tekrar eden işlemleri kolaylaştırmak için kullanılır.

View Oluşturma (SQL Sorgusu):

```
CREATE VIEW vw_MusteriArabaDetay AS

SELECT

    M.Ad AS MusteriAd,

    M.Soyad AS MusteriSoyad,

    M.Mail AS MusteriMail,

    A.Marka AS ArabaMarka,

    A.Modeli AS ArabaModeli,

    A.Üretim_Yılı AS UretimYili

FROM Musteri M

JOIN MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi

JOIN ArabaModeli A ON MA.Arabas_Modeli = A.id;

SELECT * FROM vw_MusteriArabaDetay;
```

Çıktı:

	MusteriAd	MusteriSoyad	MusteriMail	ArabaMarka	ArabaModeli	UretimYili
1	Mehmet	Demir	mehmet.demir@mail.com	Ford	Focus	2019
2	Leyla	Çakır	leyla.cakir@mail.com	Renault	Megane	2021
3	Can	Arslan	can.arslan@mail.com	Mercedes	C200	2020
4	Ali	Kaya	mehmet.kaya@mail.com	Audi	A4	2019
5	Ali	Kaya	mehmet.kaya@mail.com	Audi	A4	2019
6	Murat	Öztürk	murat.ozturk@mail.com	Peugeot	308	2020
7	Elif	Şahin	elif.sahin@mail.com	Audi	A4	2019
8	Ali	Kaya	mehmet.kaya@mail.com	Toyota	Corolla	2020
9	Zeynep	Aydın	zeynep.aydin@mail.com	Hyundai	Elantra	2021
10	Fatma	Çelik	fatma.celik@mail.com	BMW	320i	2018
11	Osman	Kurt	osman.kurt@mail.com	Volkswagen	Golf	2022
12	Ayşe	Yılmaz	ayse.yilmaz@mail.com	Honda	Civic	2021
13	Can	Arslan	can.arslan@mail.com	Audi	A4	2019
14	Can	Arslan	can.arslan@mail.com	Honda	Civic	2021

Stored Procedure Oluşturma (SQL Sorgusu):

CREATE PROCEDURE sp_MuayeneDetayByTarih

@BaslangicTarihi DATE

AS

BEGIN

SELECT

M.Ad AS MusteriAd,

M.Soyad AS MusteriSoyad,

MA.Plaka,

Mu.Tarih AS MuayeneTarihi,

Mu.Servis_Teknikeri

FROM

Musteri M

JOIN

MüsteriAraba MA ON M.id = MA.Ruhsat_Sahibi

JOIN

Muayene Mu ON MA.Plaka = Mu.Plaka

WHERE

Mu.Tarih > @BaslangicTarihi;

END;

EXEC sp_MuayeneDetayByTarih @BaslangicTarihi = '2024-01-01';

Çıktı:

109 %					
Results Messages					
	MusteriAd	MusteriSoyad	Plaka	MuayeneTarihi	Servis_Teknikeri
1	Ali	Kaya	34ABC123	2024-01-15	4
2	Ayşe	Yılmaz	35DEF456	2024-02-20	5
3	Mehmet	Demir	06GHI789	2024-03-10	6
4	Fatma	Çelik	34JKL101	2024-04-05	7
5	Can	Arslan	07MNO234	2024-05-12	8
6	Elif	Şahin	16PQR567	2024-06-20	4
7	Osman	Kurt	34STU890	2024-07-30	5
8	Leyla	Çakır	06VWX112	2024-08-15	6
9	Murat	Öztürk	07YZA345	2024-09-25	7
10	Zeynep	Aydın	34BCD678	2024-10-10	8
11	Can	Arslan	38PTR534	2024-06-19	7
12	Can	Arslan	44FZR534	2024-02-19	4